Microsoft®

Windows®

Operating System

IP Telephony with TAPI 3.0

White Paper

Abstract

TAPI 3.0 is an evolutionary API providing convergence of both traditional PSTN telephony and IP Telephony.

IP Telephony is an emerging set of technologies which enables voice, data, and video collaboration over existing LANs, WANs and the Internet.

TAPI 3.0 enables IP Telephony on the Microsoft® Windows® operating system platform by providing simple and generic methods for making connections between two or more machines, and accessing any media streams involved in the connection.

TAPI 3.0 supports standards based H.323 conferencing and IP multicast conferencing. It utilizes the Windows NT® 5.0 operating system's Active Directory service to simplify deployment within an organization, and includes quality of service (QoS) support to improve conference quality and network manageability.

© 1997 Microsoft Corporation. All rights reserved.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Microsoft, ActiveX, the BackOffice logo, DirectShow, Visual Basic, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Java is a trademark of Sun Microsystems, Inc.

Other product or company names mentioned herein may be the trademarks of their respective owners.

Microsoft Corporation • One Microsoft Way • Redmond, WA 98052-6399 • USA

0997

CONTENTS

IP TELEPHONY....2
What is IP Telephony? 2
What are the Benefits of IP Telephony? 2

INTRODUCTION TO TAPI 3.0....2
What is TAPI 3.0? 2
Inside TAPI 3.0 2
Call Control Model 2
Using TAPI Objects 2
Media Streaming Model 2

H.323 COMMUNICATIONS IN TAPI 3.0....2
What is H.323? 2
The TAPI 3.0 H.323 Telephony Service Provider 2
Integration with the Windows NT 5.0 Active Directory 2

IP MULTICAST CONFERENCING IN TAPI 3.0....2
What is IP Multicast Conferencing? 2
The TAPI 3.0 IP Multicast Conferencing TSP 2
Integration with the Windows NT 5.0 Active Directory 2
The TAPI 3.0 Rendezvous Controls 2
Conference Security Model 2

QUALITY OF SERVICE....2
What is Quality of Service? 2
Quality of Service and TAPI 3.0 2
RSVP 2
Local Traffic Control 2
IP Type of Service 2

ENTERPRISE DEPLOYMENT OF TAPI 3.0 IP TELEPHONY INFRASTRUCTURE....2
Enterprise Layout for TAPI 3.0 IP Telephony 2
Windows NT 5.0 Active Directory Layout for IP Telephony 2

TAPI 3.0 AND NETMEETING 2.0....2
What is Microsoft NetMeeting 2.0? 2
Features of TAPI 3.0 and NetMeeting 2.0 2
When to Use TAPI 3.0 and NetMeeting 2.0 2

IP TELEPHONY

What is IP Telephony?

IP Telephony is an emerging set of technologies that enables voice, data, and video collaboration over existing IP-based LANs, WANs, and the Internet.

Specifically, IP Telephony uses open IETF and ITU standards to move multimedia traffic over any network that uses IP (the Internet Protocol)—offering users both flexibility in physical media (for example, POTS lines, ADSL, ISDN, leased lines, coaxial cable, satellite, and twisted pair) and flexibility of physical location. As a result, the same ubiquitous networks that carry Web, e-mail and data traffic can be used to connect to individuals, businesses, schools and governments worldwide.

TAPI 3.0 is an evolutionary API that supports convergence of both traditional PSTN telephony and telephony over IP networks.

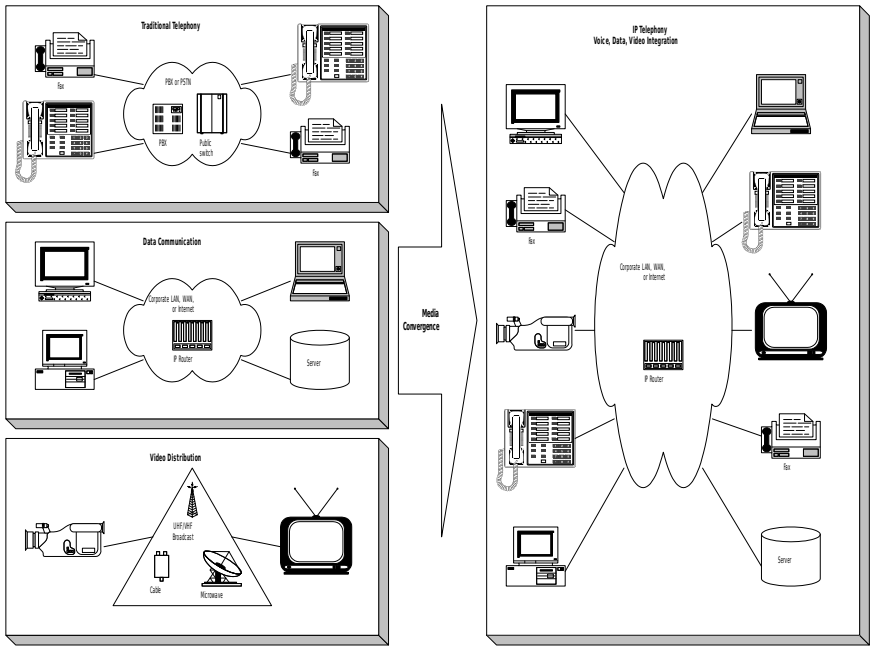
What are the Benefits of IP Telephony?

IP Telephony allows organizations and individuals to lower the costs of existing services, such as voice and broadcast video, while at the same time broadening their means of communication to include modern video conferencing, application sharing, and whiteboarding tools.

In the past, organizations have deployed separate networks to handle traditional voice, data, and video traffic. Each with different transport requirements, these networks were expensive to install, maintain, and reconfigure. Furthermore, since these networks were physically distinct, integration was difficult if not impossible, limiting their potential usefulness.

IP Telephony blends voice, video and data by specifying a common transport, IP, for each, effectively collapsing three networks into one. The result is increased manageability, lower support costs, a new breed of collaboration tools, and increased productivity.

Possible applications for IP Telephony include telecommuting, real-time document collaboration, distance learning, employee training, video conferencing, video mail, and video on demand.



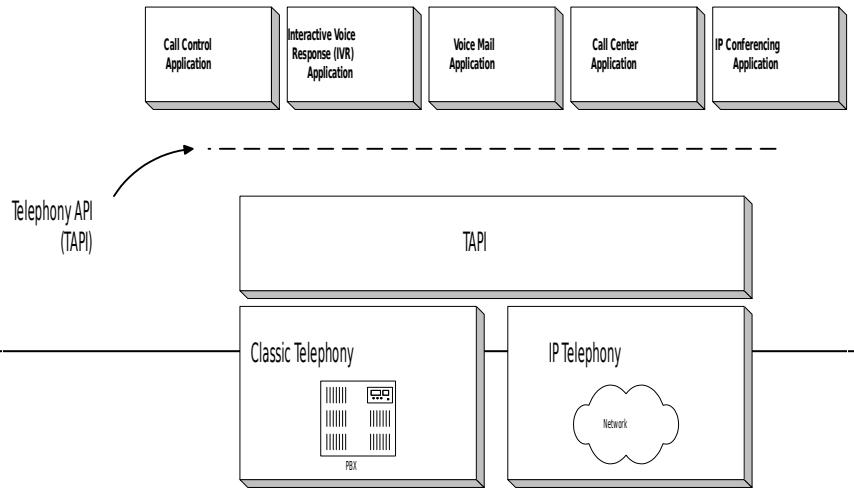
Media Convergence: Voice, Data, and Video

What is TAPI 3.0?

As telephony and call control become more common in the desktop computer, a general telephony interface is needed to enable applications to access all the telephony options available on any machine. Additionally, it is imperative that the media or data on a call is available to applications in a standard manner.

TAPI 3.0 is an architecture that provides simple and generic methods for making connections between two or more machines, and accessing any media streams involved in that connection. It abstracts call-control functionality to allow different, and seemingly incompatible, communication protocols to expose a common interface to applications.

IP Telephony is a demand poised for explosive growth, as organizations begin an historic shift from expensive and inflexible circuit-switched public telephone networks to intelligent, flexible and inexpensive IP networks. Microsoft, in anticipation of this trend, has created a robust computer telephony infrastructure, TAPI. Now in its third major version,



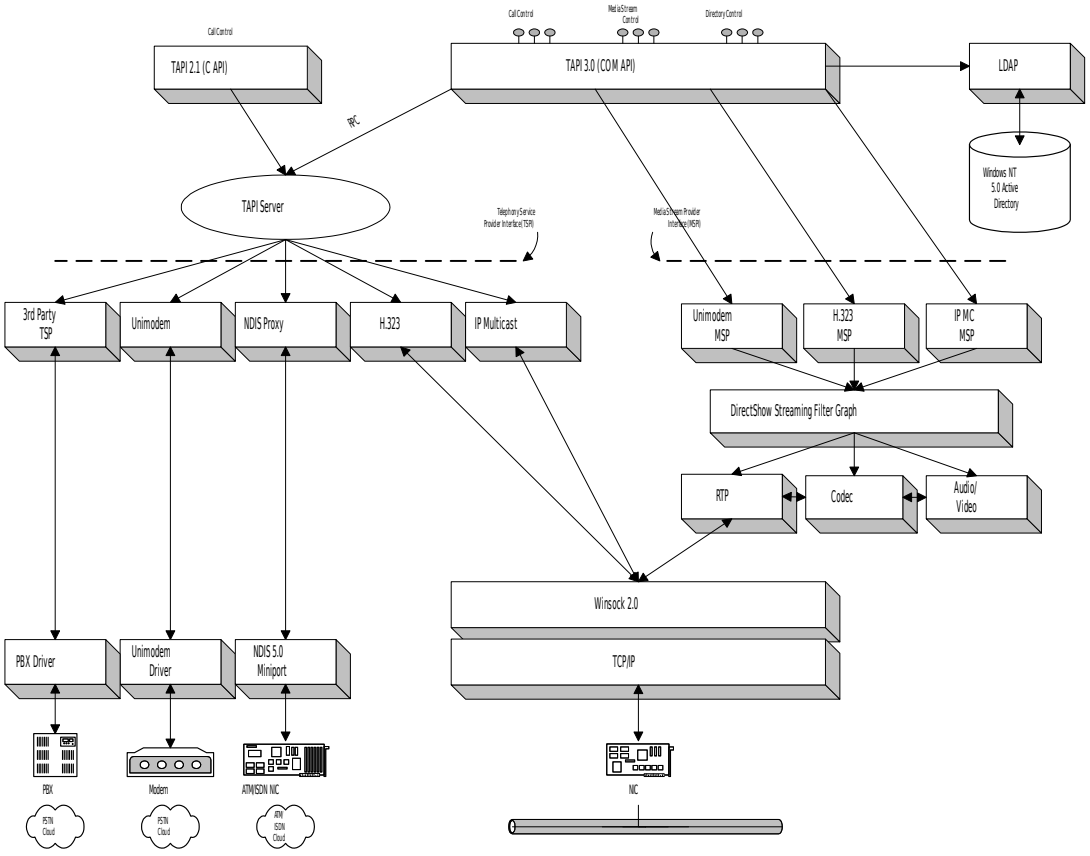
TAPI is suitable for quick and easy development of IP Telephony applications.

Convergence of IP and PSTN Telephony

Inside TAPI 3.0

TAPI 3.0 integrates multimedia stream control with legacy telephony. Additionally, it is an evolution of the TAPI 2.1 API to the COM model, allowing TAPI applications to be written in any language, such as Java™, C/C++ and the Microsoft® Visual Basic® programming system.

Besides supporting classic telephony providers, TAPI 3.0 supports standard H.323 conferencing and IP multicast conferencing. TAPI 3.0 utilizes the Windows NT® 5.0 Active Directory service to simplify deployment within an organization, and it supports quality of service (QoS) features to improve conference quality and network manageability.



TAPI Architectural Diagram

There are four major components to TAPI 3.0:

- TAPI 3.0 COM API
- TAPI Server
- Telephony Service Providers
- Media Stream Providers

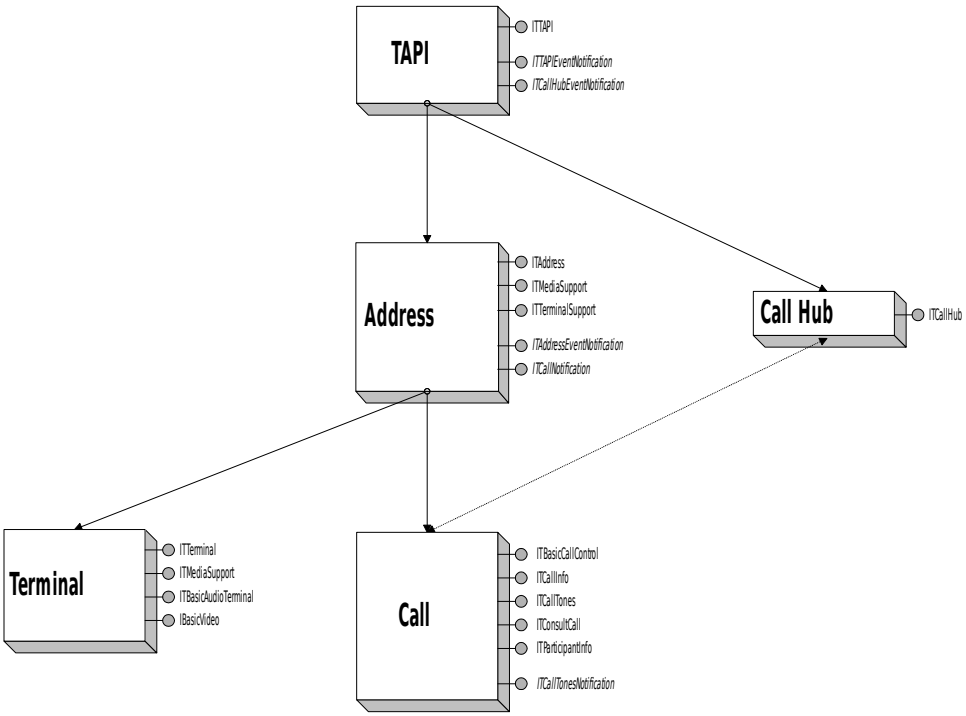
In contrast to TAPI 2.1, the TAPI 3.0 API is implemented as a suite of Component Object Model (COM) objects. Moving TAPI to the object-oriented COM model allows component upgrades of TAPI features. It also allows developers to write TAPI-enabled applications in any language, such as Java, Visual Basic, or C/C++.

The TAPI Server process (TAPISRV.EXE) abstracts the TSPI (TAPI Service Provider Interface) from TAPI 3.0 and TAPI 2.1, allowing TAPI 2.1 Telephony Service Providers to be used with TAPI 3.0, maintaining the internal state of TAPI.

Telephony Service Providers (TSPs) are responsible for resolving the protocol-independent call model of TAPI into protocol-specific call control mechanisms. TAPI 3.0 provides backward compatibility with TAPI 2.1 TSPs. Two IP Telephony service providers (and their associated MSPs) ship by default with TAPI 3.0: the H.323 TSP and the IP Multicast Conferencing TSP, which are discussed later in this document.

TAPI 3.0 provides a uniform way to access the media streams in a call, supporting the DirectShow™ API as the primary media stream handler. TAPI Media Stream Providers (MSPs) implement DirectShow interfaces for a particular TSP and are required for any telephony service that makes use of DirectShow streaming. Generic streams are handled by the application.

Call Control Model



TAPI 3.0 Object Relationships

There are five objects in the TAPI 3.0 API:

- TAPI
- Address
- Terminal
- Call
- CallHub

The *TAPI* object is the application's entry point to TAPI 3.0. This object represents all telephony resources to which the local computer has access, allowing an application to enumerate all local and remote addresses.

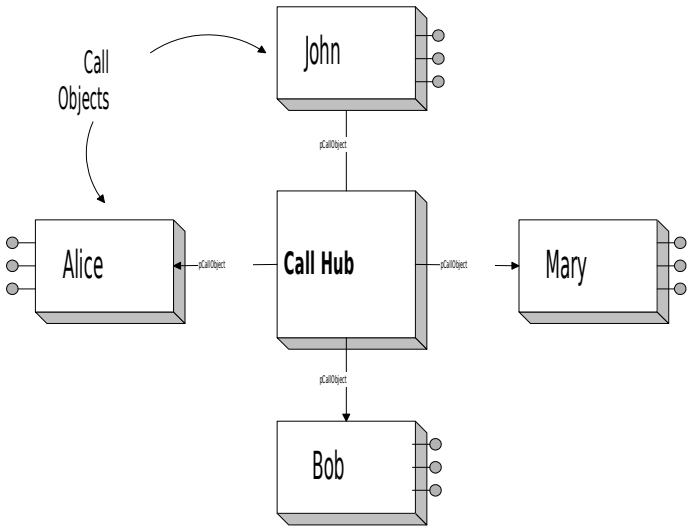
An *Address* object represents the origination or destination point for a call. Address capabilities, such as media and terminal support, can be retrieved from this object. An application can wait for a call on an Address object, or can create an outgoing call object from an Address object.

A *Terminal* object represents the sink, or renderer, at the termination or origination point of a connection. The Terminal object can map to hardware used for human interaction, such as a telephone or microphone, but can also be a file or any other device capable of receiving input or creating output.

The *Call* object represents an address's connection between the local address and one or more other addresses (This connection can be made

directly or through a CallHub). The Call object can be imagined as a first-party view of a telephone call. All call control is done through the Call object. There is a call object for each member of a CallHub.

The *CallHub* object represents a set of related calls. A CallHub object cannot be created directly by an application—they are created indirectly when incoming calls are received through TAPI 3.0. Using a CallHub object, a user can enumerate the other participants in a call or conference, and possibly (because of the location independent nature of COM) perform call control on the remote Call objects associated with those users, subject to sufficient permissions:



Call and CallHub Object Relationships

Using TAPI Objects

To Place a Call:

- Create and initialize a TAPI object.
- Use the TAPI object to enumerate all available Address objects on a machine (for example, network cards, modems, and ISDN lines).
- Enumerate the supported address types of each Address object (for example, a phone number, IP address, and so on).
- Choose an Address object based on queries for support for appropriate media (audio, video, and so on) and address types.
- Use the CreateCall() method of the Address object to create a Call object associated with a particular address.
- Select appropriate Terminals on the Call object.
- Call the Connect() method of the Call object to place the call.

To Answer a Call:

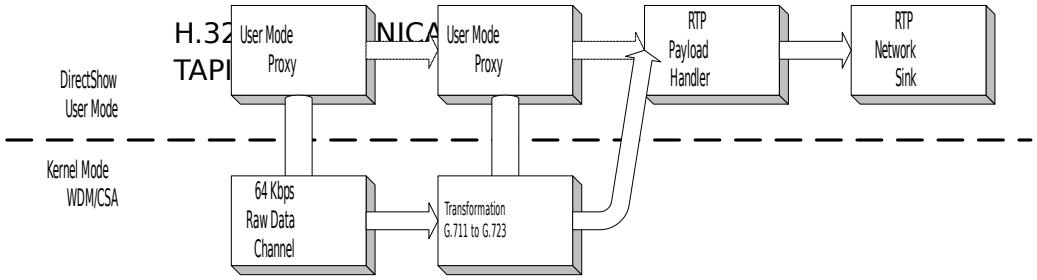
- Create and initialize a TAPI object.
- Use the TAPI object to enumerate all available Address objects on a machine (for example, network cards, modems, and ISDN lines).
- Enumerate the supported address types of each Address object (e.g. a phone number, IP address, etc.).
- Choose an Address object based on queries for support of appropriate media (audio, video, and so on) and address types.
- Register an interest in specific media types with the appropriate Address object.
- Register a call event handler (i.e. implement an ITCallNotification interface) with the Address object.
- TAPI notifies the application of a new call through ITCallNotification and creates a Call object.
- Select appropriate Terminals on the Call object.
- Call the Connect() method of the Call object to place the call.
- Call the Answer() method of the Call object to answer the call.

Media Streaming Model

The Windows® operating system provides an extensible framework for efficient control and manipulation of streaming media called the DirectShow API. DirectShow, through its exposed COM interfaces, provides TAPI 3.0 with unified stream control.

At the heart of the DirectShow services is a modular system of pluggable components called filters, arranged in a configuration called a filter graph. A component called the filter graph manager oversees the connection of these filters and controls the stream's data flow. Each filter’s capabilities are described by a number of special COM interfaces called pins. Each pin instance can consume or produce streaming data, such as digital audio.

While COM objects are usually exposed in user mode programs, the DirectShow streaming architecture includes an extension to the Windows driver model that allows the connection of media streams directly at the device driver level. The diagram below shows a simple PSTN-to-IP bridge: A 64 Kbps voice stream from an ISDN line is compressed into a G.723 audio stream and passed to an RTP payload handler, to be sent out over the network.



Sample DirectShow Filter Graph with User and Kernel Mode Components

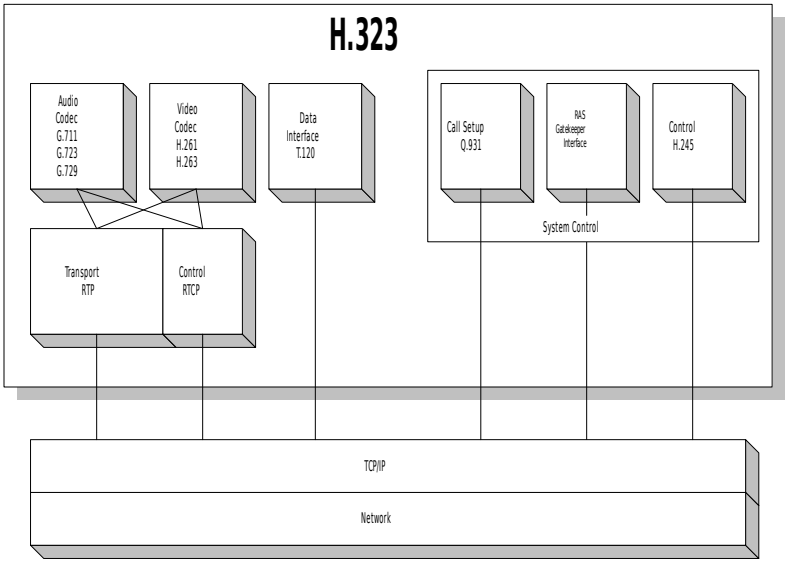
These high-performance streaming extensions to the Windows driver model avoid user-to-kernel mode transitions, and allow efficient routing of data streams between different hardware components at the device driver level. Each kernel mode filter is mirrored by a corresponding user mode proxy that facilitates connection setup and can be used to control hardware-specific features.

DirectShow network filters extend the streaming architecture to machines connected on an IP network. The Real-Time Transport protocol (RTP), designed to carry real-time data over connectionless networks, transports TAPI media streams and provides appropriate time stamp information. TAPI 3.0 includes a kernel mode RTP network filter.

TAPI 3.0 utilizes this technology to present a unified access method for the media streams in multimedia calls. Applications can route these streams by manipulating corresponding filter graphs; they can also easily connect streams from multiple calls for bridging and conferencing capabilities.

What is H.323?

H.323 is a comprehensive International Telecommunications Union (ITU) standard for multimedia communications (voice, video, and data) over connectionless networks that do not provide a guaranteed quality of service, such as IP-based networks and the Internet. It provides for call control, multimedia management, and bandwidth management for point-to-point and multipoint conferences. H.323 mandates support for standard audio and video codecs and supports data sharing via the T.120 standard. Furthermore, the H.323 standard is network, platform, and application independent, allowing any H.323 compliant terminal to interoperate with any other.



H.323 Architectural Diagram

H.323 allows multimedia streaming over current packet-switched networks. To counter the effects of LAN latency, H.323 uses as a transport the Real-time Transport Protocol (RTP), an IETF standard designed to handle the requirements of streaming real-time audio and video over the Internet.

The H.323 standard specifies three command and control protocols:

- H.245 for call control
- Q.931 for call signaling
- The RAS (Registration, Admissions, and Status) signaling function

The H.245 control channel is responsible for control messages governing operation of the H.323 terminal, including capability exchanges, commands, and indications. Q.931 is used to set up a connection between two terminals, while RAS governs registration, admission, and bandwidth functions between endpoints and gatekeepers (RAS is not used if a gatekeeper is not present). See below for more information on gatekeepers.

H.323 defines four major components for an H.323-based communications system:

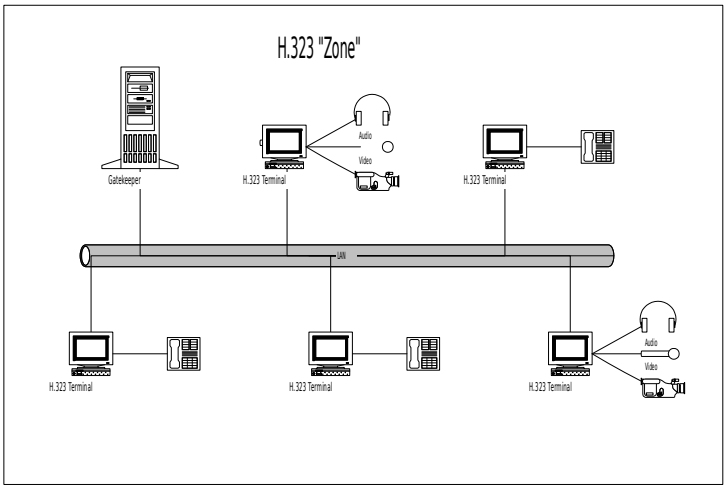
- Terminals
- Gateways
- Gatekeepers
- Multipoint Control Units (MCUs)

Terminals are the client endpoints on the network. All terminals must support voice communications; video and data support is optional.

A *Gateway* is an optional element in an H.323 conference. Gateways

bridge H.323 conferences to other networks, communications protocols, and multimedia formats. Gateways are not required if connections to other networks or non-H.323 compliant terminals are not needed.

Gatekeepers perform two important functions which help maintain the robustness of the network - address translation and bandwidth management. Gatekeepers map LAN aliases to IP addresses and provide address lookups when needed. Gatekeepers also exercise call control functions to limit the number of H.323 connections, and the total bandwidth used by these connections, in an H.323 “zone.” A Gatekeeper is not required in an H.323 system—however, if a Gatekeeper is present, terminals must make use of its services.



H.323 Components

Multipoint Control Units (MCU) support conferences between three or more endpoints. An MCU consists of a required Multipoint Controller (MC) and zero or more Multipoint Processors (MPs). The MC performs H.245 negotiations between all terminals to determine common audio and video processing capabilities, while the Multipoint Processor (MP) routes audio, video, and data streams between terminal endpoints.

Any H.323 client is guaranteed to support the following standards: H.261 and G.711. H.261 is an ITU-standard video codec designed to transmit compressed video at a rate of 64 Kbps and at a resolution of 176x44 pixels (QCIF). G.711 is an ITU-standard audio codec designed to transmit A-law and µ-law PCM audio at bit rates of 48, 56, and 64 Kbps.

Optionally, an H.323 client may support additional codecs: H.263 and G.723. H.263 is an ITU-standard video codec based on and compatible with H.261. It offers improved compression over H.261 and transmits video at a resolution of 176 x 44 pixels (QCIF). G.723 is an ITU-standard

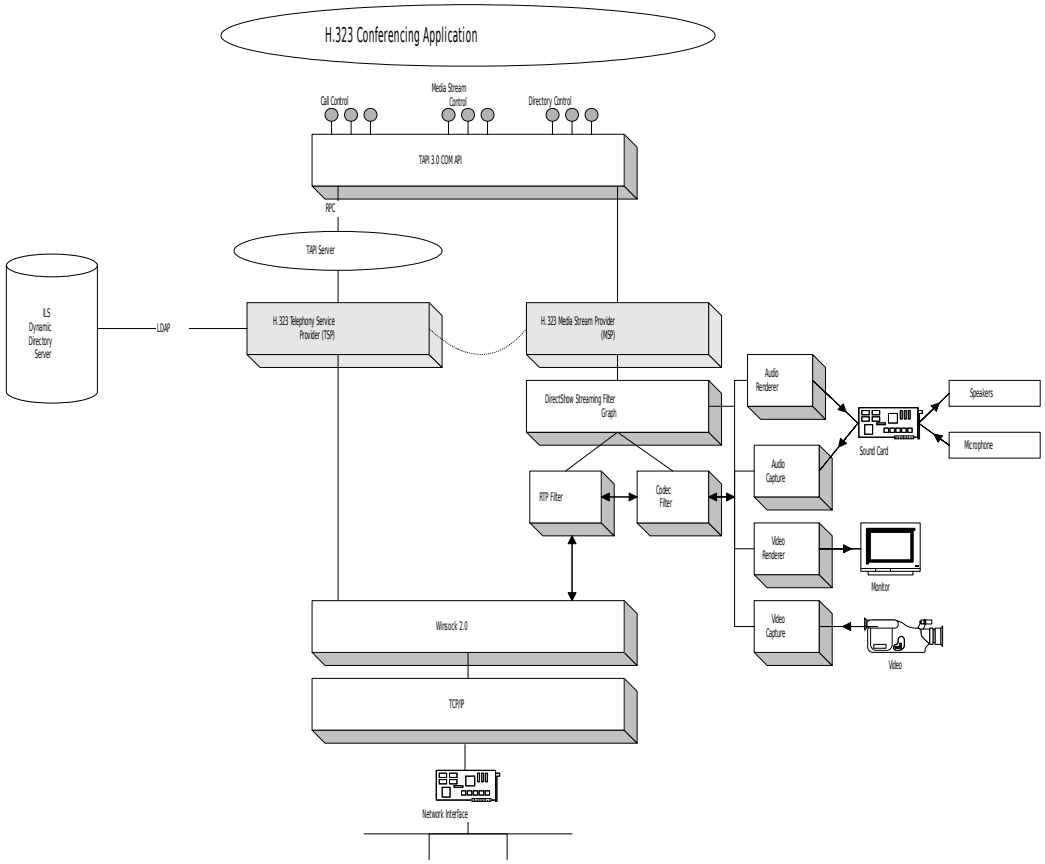
audio codec designed to operate at very low bit rates.

The TAPI 3.0 H.323 Telephony Service Provider

The H.323 Telephony Service Provider (along with its associated Media Stream Provider) allows TAPI-enabled applications to engage in multimedia sessions with any H.323-compliant terminal on the local area network.

Specifically, the H.323 Telephony Service Provider (TSP) implements the H.323 signaling stack. The TSP accepts a number of different address formats, including name, machine name, and e-mail address.

The H.323 MSP is responsible for constructing the DirectShow filter graph for an H.323 connection (including the RTP, RTP payload handler, codec, sink, and renderer filters).



H.323 TSP Architectural Diagram

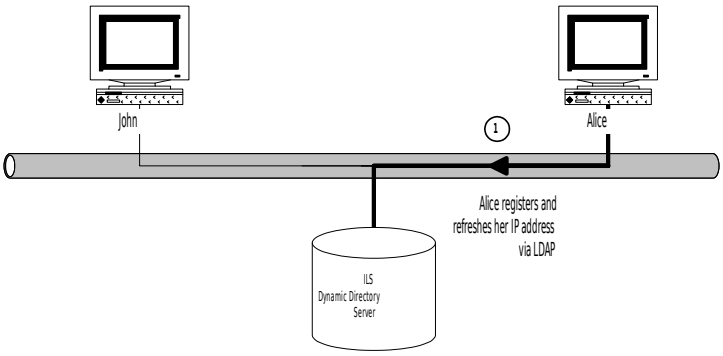
Integration with the Windows NT 5.0 Active Directory

H.323 telephony is complicated by the reality that a user's network address (in this case, a user's IP address) is highly volatile and cannot be counted on to remain unchanged between H.323 sessions. The TAPI

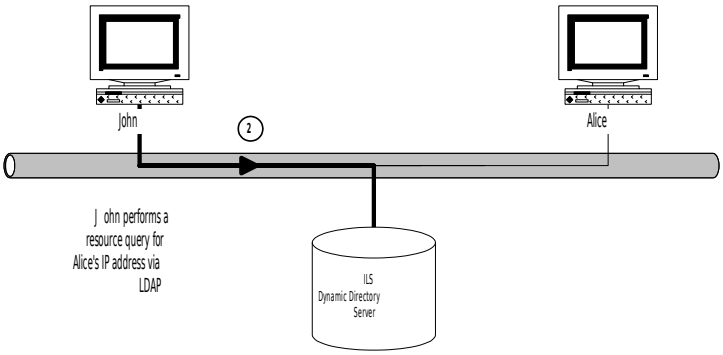
H.323 TSP utilizes the services of the Windows NT Active Directory to perform user-to-IP address resolution. Specifically, user-to-IP mapping information is stored and continually refreshed using the Internet Locator Service (ILS) Dynamic Directory, a real-time server component of the Active Directory.

The following user scenario illustrates IP address resolution in the H.323 TSP:

1. John wishes to initiate an H.323 conference with Alice, another user on the LAN. Once Alice's video conferencing application creates an Address object and puts it in listen mode, Alice's IP address is added to the Windows NT Active Directory by the H.323 TSP. This information has a finite time to live (TTL) and is refreshed at regular intervals via the Lightweight Directory Access Protocol (LDAP):

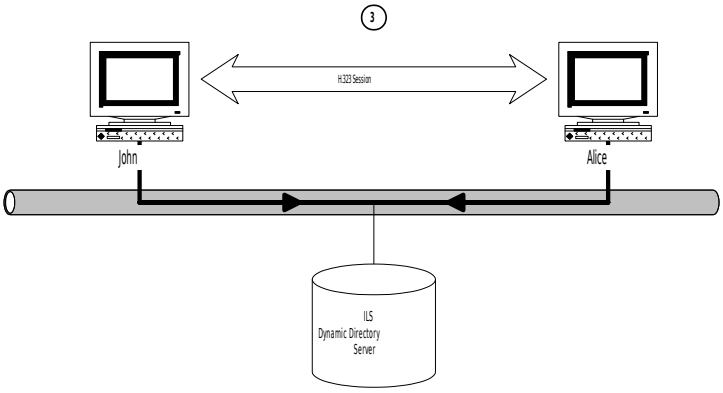


2. John's H.323 TSP then queries the ILS Dynamic Directory server for Alice's IP address. Specifically, John queries for any and all RTPerson objects in the Directory associated with Alice:



3. Armed with Alice's up-to-date IP address, John initiates an H.323 call to Alice's machine, and H.323-standard negotiations and media selection occurs between the peer TSPs on both machines. Once capability negotiations have been completed, both H.323 Media Stream Providers

(MSPs) construct appropriate DirectShow filter graphs, and all media streams are passed off to DirectShow to handle. The conference then begins.



What is IP Multicast Conferencing?

IP Multicast is an extension to IP that allows for efficient group communication. IP Multicast arose out of the need for a lightweight, scalable conferencing solution that solved the problems associated with real-time traffic over a datagram, “best-effort” network. There are many advantages to using IP Multicast: scalability, fault tolerance, robustness, and ease of setup.

The IP Multicast conferencing model incorporates the following key features:

No global coordination is needed to add and remove members from a conference.

To reach a multicast group, a user sends data to a single multicast IP address. No knowledge of the other users in a group is necessary.

To receive data, users register their interest in a particular multicast IP address with a multicast aware router. No knowledge of the other users in a group is necessary.

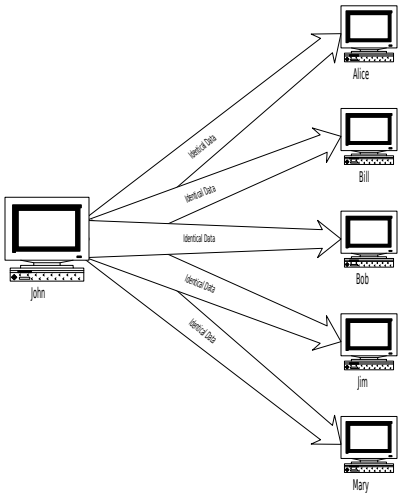
Routers hide the multicast implementation details from the user.

Traditional connection-oriented conferencing suffers from a number of problems:

User complexity: Users must know the location of every user they wish to converse with, limiting scalability and fault-tolerance and rendering it difficult for users to add and remove themselves from a conference.

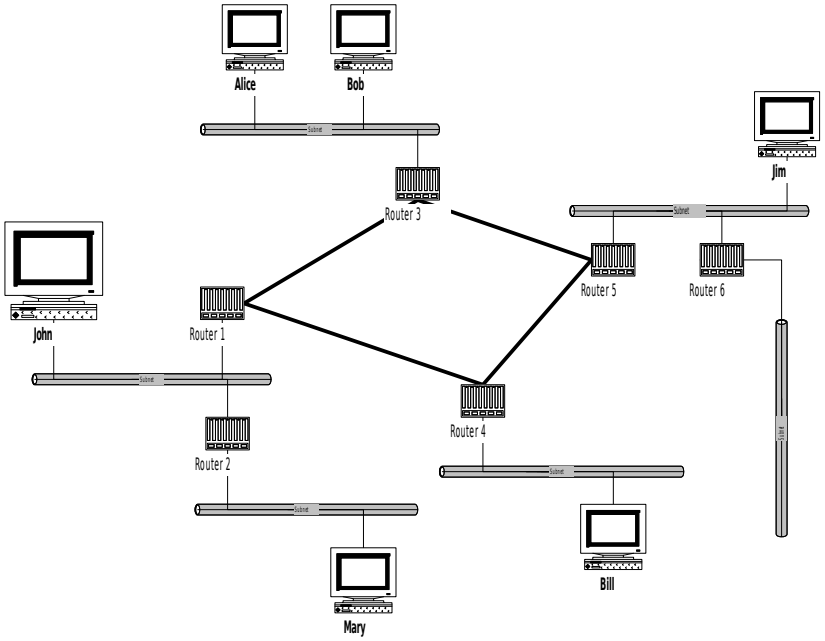
Wasted bandwidth: A user wishing to broadcast data to *N* users must

send data through *N* connections, as shown in the following diagram:



Network Topology: Sender's View

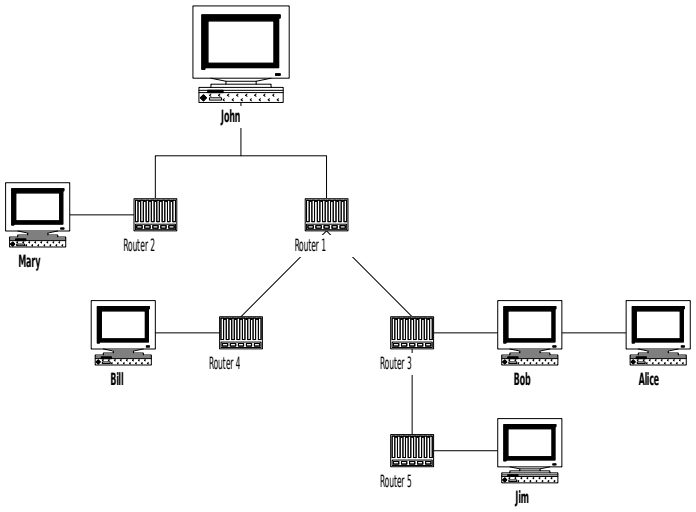
The total bandwidth required for multiparty conferences in which all users are sending data goes up as N^2 the number of parties involved, leading to huge scalability problems. IP Multicast takes advantage of the actual network topology to eliminate the transmission of redundant data down the same communications links.



Actual Network Topology

IP Multicast implements a lightweight, session-based communications model, which places relatively little burden on conference users. Using IP Multicast, users send only one copy of their information to a group IP address that reaches all recipients. IP Multicast is designed to scale well as the number of participants expands—adding one more user does not add a corresponding amount of bandwidth. Multicasting also results in a greatly reduced load on the sending server.

IP Multicast routes these one-to-many data streams efficiently by constructing a spanning tree, in which there is only one path from one router to any other. Copies of the stream are made only when paths diverge:



IP Multicast Utilizing a Spanning Tree

Without multicasting, the same information must either be carried over the network multiple times, one time for each recipient, or broadcast to everyone on the network, consuming unnecessary bandwidth and processing.

IP Multicast uses Class D Internet Protocol addresses to specify multicast host groups, ranging from 224.0.0.0 to 239.255.255.255. Both permanent and temporary group addresses are supported. Permanent addresses are assigned by the Internet Assigned Numbers Authority (IANA) and include 224.0.0.1, the "all-hosts group" used to address all multicast hosts on the local network, and 224.0.0.2, which addresses all routers on a LAN. The range of addresses between 224.0.0.0 and 224.0.0.255 is reserved for routing and other low-level network protocols. Other addresses and ranges have been reserved for applications, such as 224.0.13.000 to 224.0.13.255 for Net News (for more information, see RFC 1700, “Assigned Numbers” at ftp://ftp.internic.net/rfc/rfc1700.txt).

The transport protocol for IP Multicast is RTP (Real-time Transport Protocol), which provides a standard multimedia header giving timestamp, sequence numbering, and payload format information.

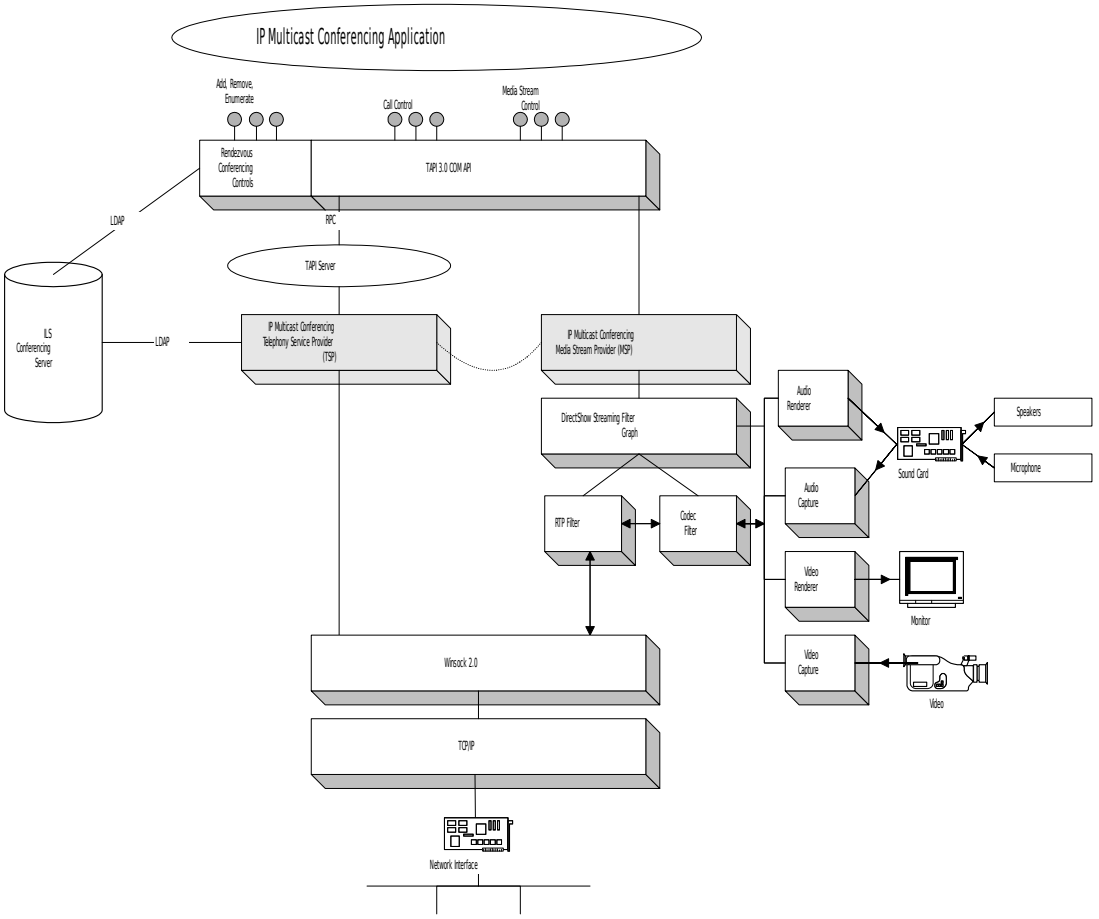
Applications for IP Multicast include video and audio conferencing, telecommuting, database and Web-site replication, distance learning, dissemination of stock quotes, and collaborative computing. At present, the largest demonstration of the capabilities of IP Multicast is the Internet MBONE (Multicast Backbone).

The MBONE is an experimental, global multicast network layered on top of the physical Internet. It has been in existence for about five years, and presently carries IETF meetings, NASA space shuttle launches, music, concerts, and many other live meetings and performances (for more information, see http://www.mbone.com).

The TAPI 3.0 IP Multicast Conferencing TSP

The IP Multicast Conferencing TSP is chiefly responsible for resolving conference names to IP multicast addresses, using the Session Description Protocol (SDP) conference descriptors stored in the ILS Dynamic Directory Conference Server. It is complemented by the Rendezvous conference controls, described later in this document.

The IP Multicast Conferencing MSP is responsible for constructing an appropriate DirectShow filter graph for an IP multicast connection (including RTP, RTP payload handler, codec, sink, and renderer filters).



Architectural Diagram

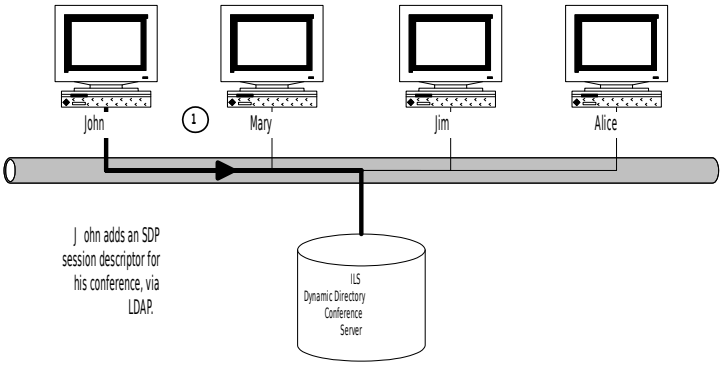
Integration with the Windows NT 5.0 Active Directory

TAPI 3.0 uses the IETF standard Session Description Protocol to advertise IP multicast conferences across the enterprise. SDP descriptors are stored in the Windows NT Active Directory—specifically, in the ILS Dynamic Directory Conference Server. SDP is discussed in more detail later in this document. In contrast to the Dynamic Directory servers utilized by the H.323 TSP, there is only one ILS Conference Server per enterprise, since conference announcements are not continually refreshed, therefore consuming little bandwidth.

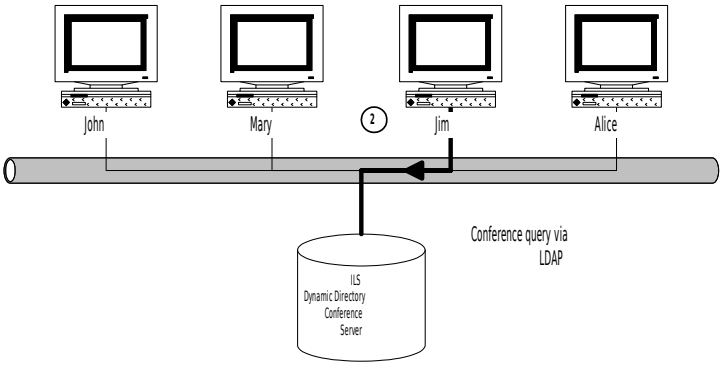
TAPI 3.0’s IP multicast conference mechanism is illustrated in the following scenario, in which John wishes to initiate a multicast conference:

1. John’s TAPI 3.0-enabled application utilizes the Rendezvous Controls (discussed in more detail later in this document) to create an SDP session descriptor on the ILS Conference Server. The SDP descriptor contains, among other things, the conference name, start and end time

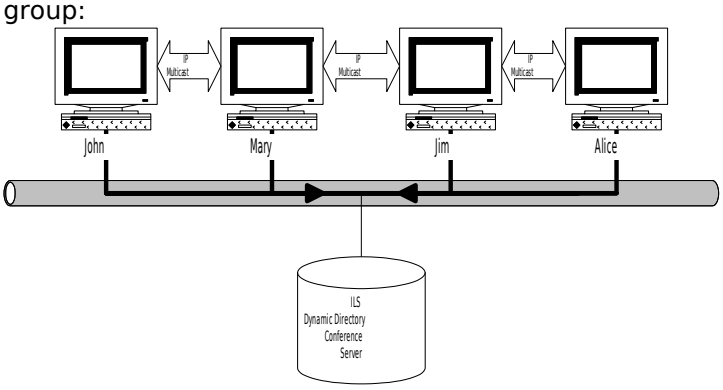
information, the IP multicast address of the conference, and the media types used for the conference.



2. Jim queries the ILS Conference Server for SDP descriptors of conferences matching his criteria:

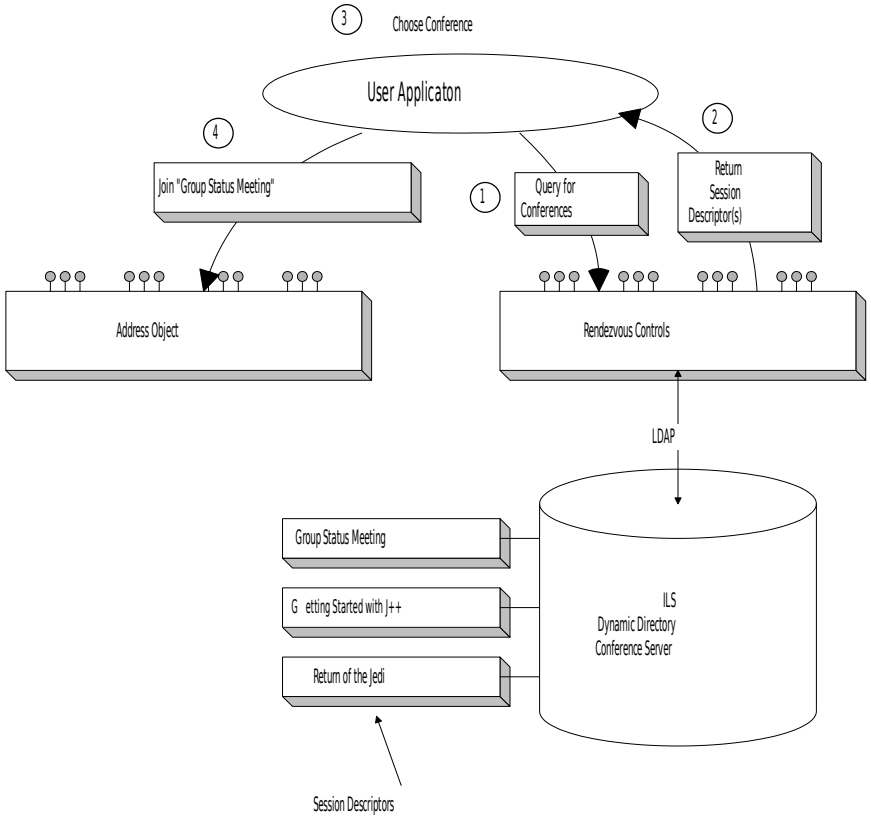


3. Mary and Alice perform similar queries and use the SDP information they receive to decide to participate in John's conference. Armed with the multicast IP address of the conference, they join the multicast host group:



The TAPI 3.0 Rendezvous Controls

The Rendezvous Controls are a set of COM components that abstract the concept of a conference directory, providing a mechanism to advertise new multicast conferences and to discover existing ones. They provide a common schema (SDP) for conference announcement, as well as scriptable interfaces, authentication, encryption, and access control features.



Joining a Conference Using the Rendezvous Controls

The user may add, delete, and enumerate multicast conferences stored on an ILS Conference Server via the Rendezvous Controls. These controls manipulate conference data via the Lightweight Directory Access Protocol (LDAP).

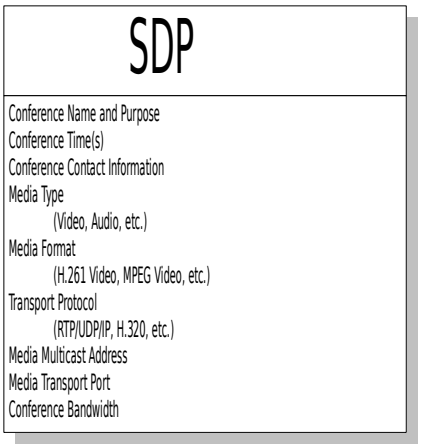
Joining a conference is illustrated in the diagram above. The conferencing application uses the Rendezvous Controls to obtain session descriptors for the conferences that match the user's criteria (1,2). Access control lists (ACLs) protect each of the stored conference announcements, and whether or not an announcement is visible and accessible depends upon the user's credentials.

Once the user has chosen a conference (3), the user application searches

for all Address objects that support the address type “Multicast Conference Name.” The application then uses the conference name from the SDP descriptor as a parameter to the CreateCall() method of the appropriate Address object (4), passes the appropriate Terminal objects to the returned Call object, and calls Call->Connect().

The Rendezvous Controls store the conference information on an ILS Conference Server in a format defined by the Session Description Protocol (SDP), an IETF standard for announcing multimedia conferences. The purpose of SDP is to publicize sufficient information about a conference (time, media, and location information) to allow prospective users to participate if they so choose. Originally designed to operate over the Internet MBONE (IP Multicast Backbone), SDP has been integrated by TAPI 3.0 with the Windows NT Active Directory, thereby extending its functionality to local area networks.

An SDP descriptor advertises the following information about a conference:



General SDP Attributes

A session description is broken into three main parts: a single Session Description, zero or more Time Descriptions, and zero or more Media Descriptions. The Session Description contains global attributes that apply to the whole conference or all media streams. Time Descriptions contain conference start, stop, and repeat time information, while Media Descriptions contain details that are specific to a particular media stream.

While traditional IP multicast conferences operating over the MBONE have advertised conferences using a push model based on the Session Announcement Protocol (SAP), TAPI 3.0 utilizes a pull-based approach using Windows NT Active Directory services. This approach offers numerous advantages, among them bandwidth conservation and ease of

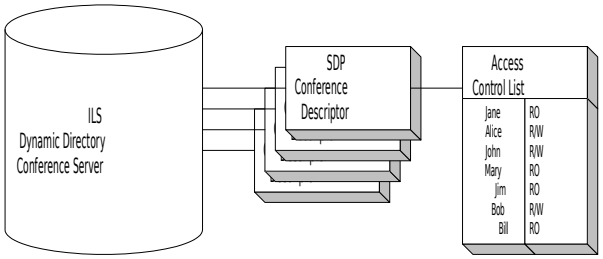
administration. See the “Integration with Windows NT 5.0 Active Directory” subsection for details.

Conference Security Model

TAPI 3.0’s conference security system addresses the following needs:

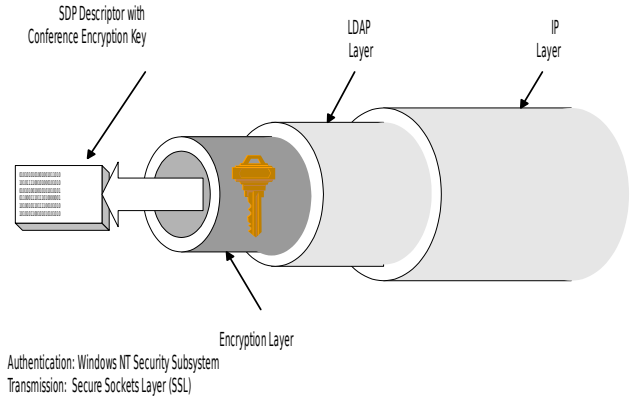
- Controlling who can create, delete, and view conference announcements.
- Preventing conference eavesdropping.

TAPI 3.0 utilizes the security features of the Windows NT Active Directory and LDAP to provide for secure conferencing over insecure networks such as the Internet. Each object in the Active Directory can be associated with an Access Control List (ACL) specifying object access rights on a user or group basis. By associating ACLs with SDP conference descriptors, conference creators can specify who can enumerate and view conference announcements. User authentication is provided by the Windows NT security subsystem.



SDPs and ACLs

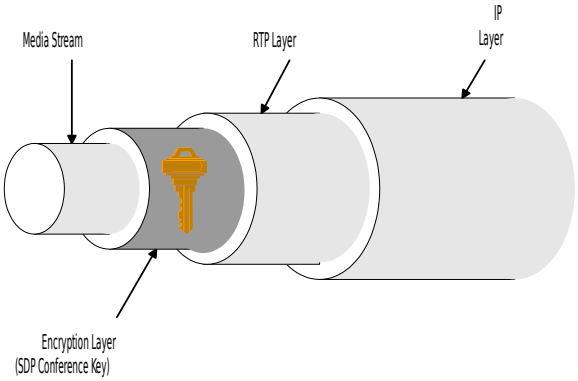
Session Descriptors are transmitted from the ILS Conference Server to the user over LDAP in encrypted form, via a Secure Sockets Layer (SSL) connection, ensuring that the SDP is safe from eavesdroppers:



Distribution of the SDP

QUALITY OF SERVICE

IP Multicast makes no provision for authenticating users - any user may anonymously join a multicast host group. To keep conferences private, TAPI 3.0 allows an IP multicast conference to be encrypted, with the encryption key distributed from within the conference descriptor. Only users with sufficient permissions have access to a conference's SDP descriptor, and therefore the Multicast Encryption Key. Once an authenticated user fetches the encryption key, he or she can participate in the conference.



Encrypted Multicast Stream

What is Quality of Service?

In contrast to traditional data traffic, multimedia streams, such as those used in IP Telephony or videoconferencing, may be extremely bandwidth and delay sensitive, imposing unique quality of service (QoS) demands on the underlying networks that carry them. Unfortunately, IP, with a connectionless, "best-effort" delivery model, does not guarantee delivery of packets in order, in a timely manner, or at all. In order to deploy real-time applications over IP networks with an acceptable level of quality, certain bandwidth, latency, and jitter requirements must be guaranteed, and must be met in a fashion that allows multimedia traffic to coexist with traditional data traffic on the same network.

Bandwidth: Multimedia data, and in particular video, may require orders of more bandwidth than traditional networks have been provisioned to handle. An uncompressed NTSC video stream, for example, can require upwards of 220 megabits per second to transmit. Even compressed, a handful of multimedia streams can completely overwhelm any other traffic on the network.

Latency: The amount of time a multimedia packet takes to get from the source to the destination (latency) has a major impact on the perceived quality of the call. There are many contributors towards latency, including transmission delays, queuing delays in network equipment, and

delays in host protocol stacks. Latency must be minimized in order to maintain a certain level of interactivity and to avoid unnatural pauses in conversation.

Jitter: In contrast to data traffic, real-time multimedia packets must arrive in order and on time to be of any use to the receiver. Variations in packet arrival time (jitter) must be below a certain threshold to avoid dropped packets (and therefore irritating shrieks and gaps in the call). Jitter, by determining receive buffer sizes, also affects latency.

Coexistence: In comparison with multimedia traffic, data traffic is relatively bursty, and arrives in unpredictable chunks (for instance, when someone opens a Web page, or downloads a file from an FTP site). Aggregations of such bursts can clog routers and cause gaps in multimedia conferences, leaving calls at the mercy of everyone on the network (including other IP Telephony users). Multimedia bandwidth must be protected from data traffic, and vice versa.

Public-switched telephone networks guarantee a minimum quality of service by allocating static circuits for every telephone call. Such an approach is simple to implement, but wastes bandwidth, lacks robustness, and makes voice, video, and data integration difficult. Furthermore, circuit-switched data paths are impossible to create using a connectionless network such as IP.

QoS support on IP networks offers the following benefits:

- Support for real-time multimedia applications.
- Assurance of timely transfers of large amounts of data.
- The ability to share the network in a manner that avoids starving applications of bandwidth.

Quality of Service and TAPI 3.0

Quality of service in TAPI 3.0 is handled through the DirectShow RTP filter, which negotiates bandwidth capabilities with the network based on the requirements of the DirectShow codecs associated with a particular media stream. These requirements are indicated to the RTP filter by the codecs via its own QoS interface. The RTP filter then uses the COM Winsock2 GQoS interfaces to indicate, in an abstract form, its QoS requirements to the Winsock2 QoS service provider (QoS SP). The QoS SP, in turn, invokes a number of varying QoS mechanisms appropriate for the application, the underlying media, and the network, in order to guarantee appropriate end-to-end QoS. These mechanisms include:

- The Resource Reservation Protocol (RSVP)
- Local Traffic Control:
  - Packet Scheduling
  - 802.1p
  - Appropriate layer 2 signaling mechanisms

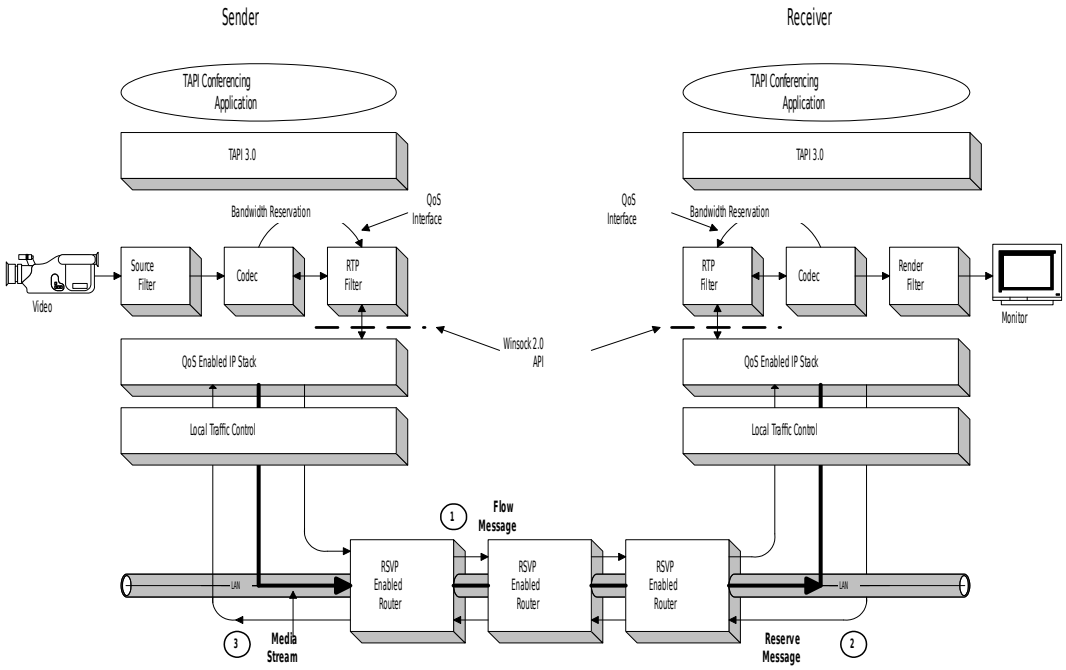
- IP Type of Service and DTR header settings

RSVP

The **Resource Reservation Protocol** (RSVP) is an IETF standard designed to support resource (for example, bandwidth) reservations through networks of varying topologies and media. Through RSVP, a user's quality of service requests are propagated to all routers along the data path, allowing the network to reconfigure itself (at all network levels) to meet the desired level of service.

The RSVP protocol engages network resources by establishing *flows* throughout the network. A *flow* is a network path associated with one or more senders, one or more receivers, and a certain quality of service. A sending host wishing to send data that requires a certain QoS will broadcast, via an RSVP-enabled Winsock Service Provider, "path" messages toward the intended recipients. These path messages, which describe the bandwidth requirements and relevant parameters of the data to be sent, are propagated to all intermediate routers along the path.

A receiving host, interested in this particular data, will confirm the flow (and the network path) by sending "reserve" messages through the network, describing the bandwidth characteristics of data it wishes to receive from the sender. As these reserve messages propagate back toward the sender, intermediate routers, based on bandwidth capacity, decide whether or not to accept the proposed reservation and commit resources. If an affirmative decision is made, the resources are committed and reserve messages are propagated to the next hop on the path from source to destination.



Resource Reservation with TAPI 3.0

Local Traffic Control

Packet Scheduling: This mechanism can be used in conjunction with RSVP (if the underlying network is RSVP-enabled) or without RSVP. Traffic is identified as belonging to one flow or another, and packets from each flow are scheduled in accordance with the traffic control parameters for the flow. These parameters generally include a scheduled rate (token bucket parameter) and some indication of priority. The former is used to pace the transmission of packets to the network. The latter is used to determine the order in which packets should be submitted to the network when congestion occurs.

801.2p: Traffic control can also be used to determine the 802.1 *User Priority* value (a MAC header field used to indicate relative packet priority) to be associated with each transmitted packet. 802.1p-enabled switches can then give preferential treatment to certain packets over others, providing additional quality of service support at the data link layer level.

Layer 2 Signaling Mechanisms: In response to Winsock 2 QoS APIs, the QoS service provider may invoke additional traffic control mechanisms depending on the specific underlying data link layer. It may signal an underlying ATM network, for instance, to set up an appropriate virtual circuit for each flow. When the underlying media is a traditional 802 shared media network, the QoS service provider may extend the standard RSVP mechanism to signal a Subnet Bandwidth Manager (SBM). The SBM provides centralized bandwidth management on shared

networks.

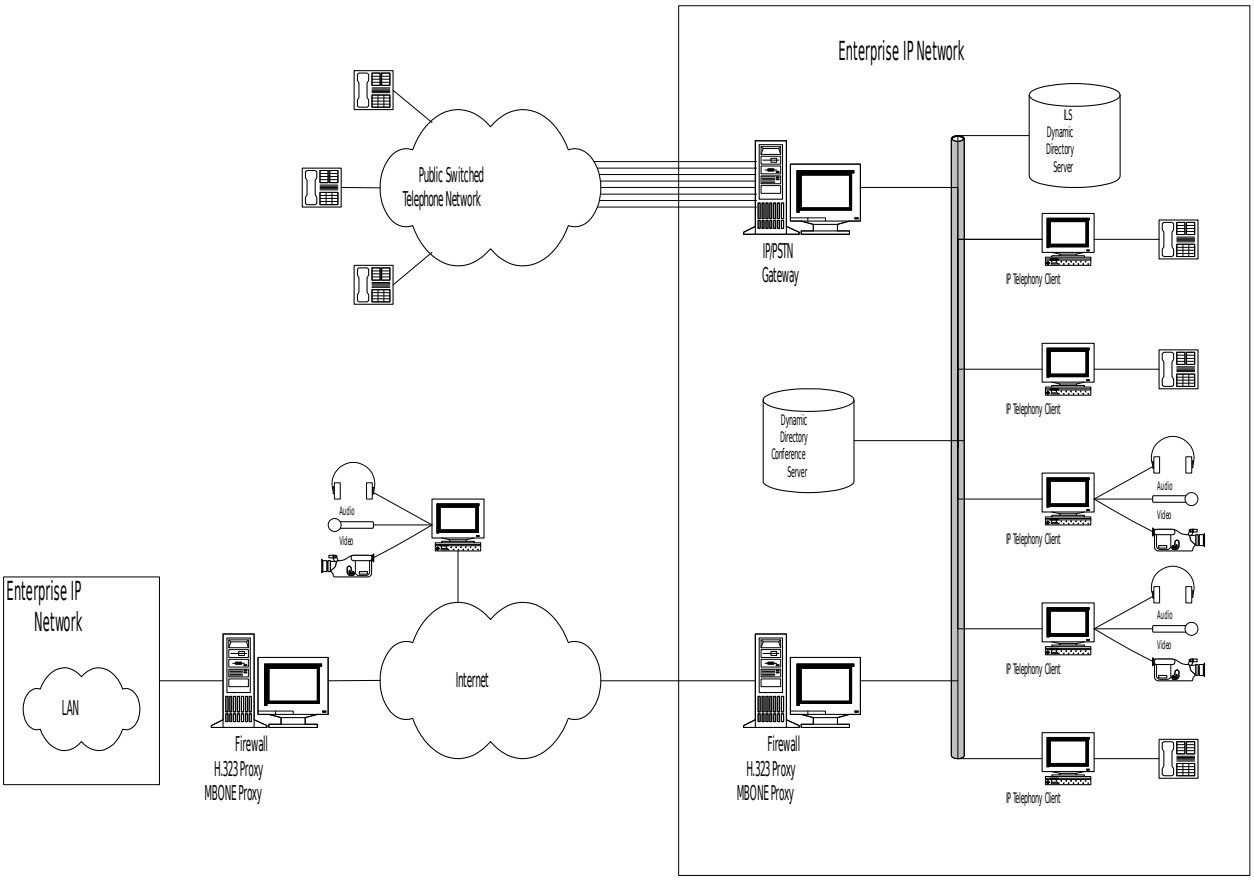
IP Type of Service

Each IP packet contains a three-bit *Precedence* field, which indicates the priority of the packet. An additional field can be used to indicate a delay, throughput, or reliability preference to the network. Local traffic control can be used to set these bits in the IP headers of packets on particular flows. As a result, packets belonging to a flow will be treated appropriately later by three devices on the network. These fields are analogous to 802.1p priority settings but are interpreted by higher layer network devices.

TAPI 3.0 has been designed to scale from the smallest business up to the largest organizations, while at the same time taking advantage of the Windows NT Active Directory to bring IP Telephony to the enterprise.

Enterprise Layout for TAPI 3.0 IP Telephony

The diagram below illustrates the enterprise layout for a sample enterprise with two sites connected through the Internet. The ILS Dynamic Directory Servers and the ILS Dynamic Directory Conference Server, as explained above, provide functionality for point-to-point and multiparty conferencing. IP Telephony clients can utilize video and audio capture equipment, but can also support legacy telephones through the use of a PSTN add-in card.



Enterprise Layout for IP Telephony

The IP/PSTN Gateway digitizes incoming analog voice calls from PSTN lines and encapsulates

them in H.323 streams, and vice versa, providing users with the ability to send and receive legacy voice calls through existing telephony infrastructure.

The H.323 Proxy allows H.323 clients connectivity with the Internet by forwarding H.323 streams through the enterprise firewall. This enables H.323 Internet, Intranet, and business-to-business connectivity.

The function of the IP Multicast Proxy is somewhat similar to that of the H.323 Proxy—to forward multicast conference packets - but also furnishes clients with the ability to propagate selected conference announcements to and from the Internet.

The IP Multicast Proxy monitors conference announcements stored on the ILS Dynamic Directory Conference Server and broadcasts conferences with appropriate scope and security attributes to the Internet using the

Session Announcement Protocol (SAP).

Conversely, the IP Multicast Proxy listens for appropriate conferences from those broadcast over the Internet and populates the ILS Dynamic Directory Conference Server with these announcements. In this manner, the IP Multicast Proxy allows users conference connectivity over the Internet while ensuring the confidentiality and security of private conferences.

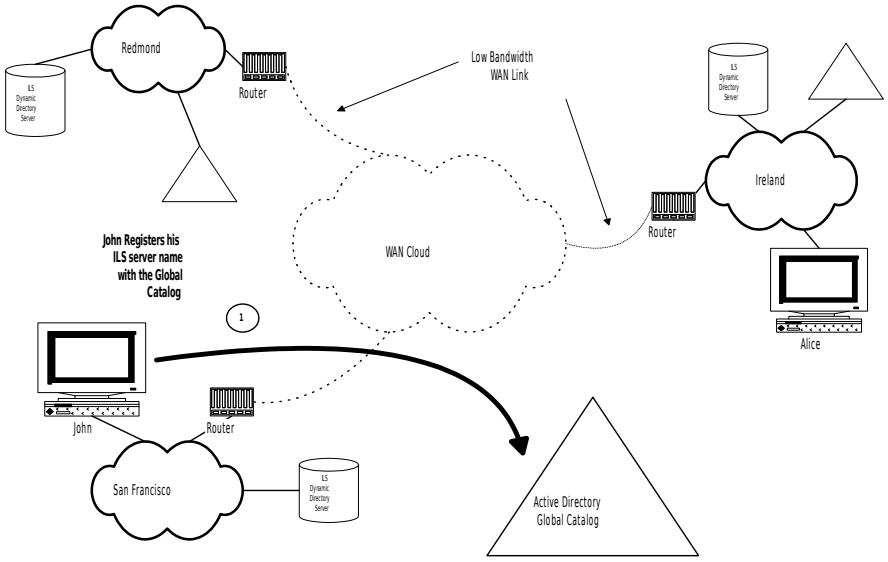
Windows NT 5.0 Active Directory Layout for IP Telephony

As discussed earlier, the H.323 TSP uses the services of the ILS Dynamic Directory component of the Active Directory to remove the burden of name to IP translation from the user.

At the network level, the Windows NT Active Directory model treats an organization as a collection of sites. Sites are regions of good connectivity, such as subnets or LANs, and typically correlate with physical locales such as campuses.

For bandwidth and performance reasons, ILS servers are typically distributed across the enterprise, one per site, with each ILS server (or a replicating cluster of severs) being responsible for maintaining user-to-IP mappings for their site. To conserve bandwidth, these volatile mappings are not replicated across sites.

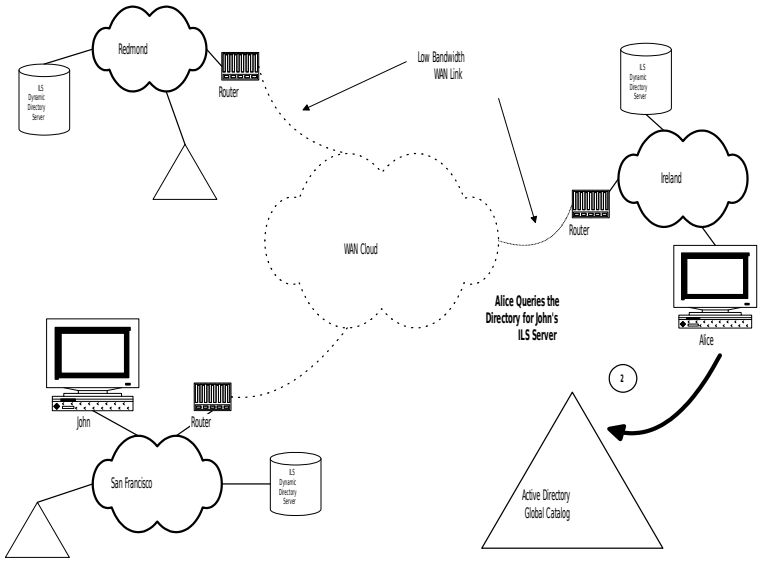
TAPI 3.0 utilizes the Active Directory to associate users with particular ILS servers. Users wishing to place an IP telephone call first consult the Global Catalog (a replicated subset of the Active Directory) for the User object of the person they wish to call. The Telephony container in the User object contains the name of the ILS server for that user’s site, which is then queried for the IP address in question.



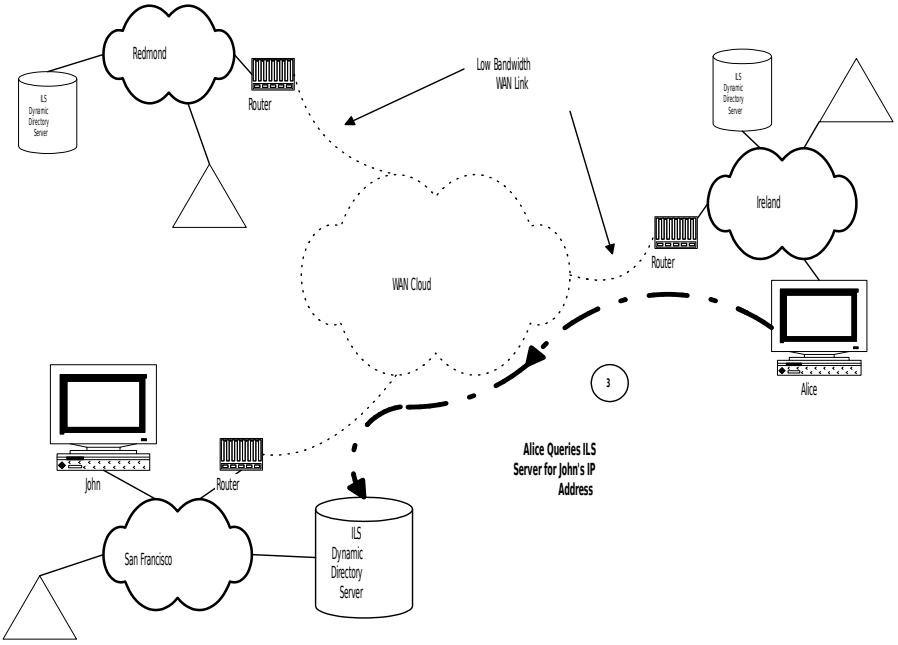
The following scenario illustrates enterprise deployment of the TAPI 3.0 directory infrastructure. In this example, Alice wishes to initiate an H.323 call to John:

John's must previously register his ILS server name with the Active Directory Global Catalog. Upon initialization, John's H.323 TSP queried the Global Catalog for the Subnet object associated with his machine, and from that has derived what site John's subnet and machine belong to. The TSP then fetches the name of the ILS server (or cluster of replicating servers) from DNS records, and stores this information in John's User object.

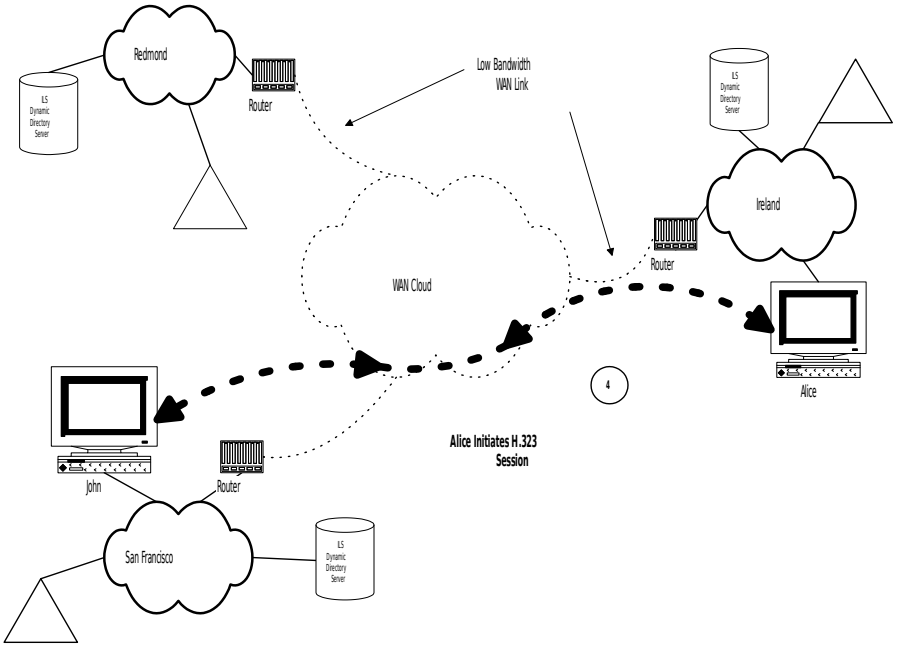
Subsequently, Alice's H.323 TSP queries her local copy of the Global Catalog for the name of John's ILS server:



Alice's H.323 TSP then queries John's ILS server across the WAN for John's current IP address:



Alice then initiates an H.323 session with John:



The call abstraction inherent in TAPI allows this ILS and Active Directory interaction to occur transparently both to the user, and to the TAPI 3.0-enabled application.

TAPI 3.0 AND NETMEETING 2.0

What is Microsoft NetMeeting 2.0?

Microsoft NetMeeting is a conferencing and collaboration tool designed for the Internet or intranet. NetMeeting also provides a set of programming interfaces for adding conferencing functionality to your applications. It helps small and large organizations take full advantage of the global reach of the Internet or corporate intranet for real-time communications and collaboration by combining IP Telephony and Conferencing functionality. Connecting to other NetMeeting users is made easy with the Microsoft Internet Locator Server (ILS), enabling participants to call each other from a dynamic directory within NetMeeting or from a World Wide Web page. While connected on the Internet or corporate intranet, participants can communicate with both voice and video, work together on virtually any Windows-based application, exchange or mark up graphics on an electronic whiteboard, transfer files, or use the text-based chat program. For more information on Microsoft NetMeeting 2.0, see http://www.microsoft.com/netmeeting.

Microsoft NetMeeting 2.0 has the following features:

H.323 standards-based voice and video conferencing. Real-time, point-to-point audio conferencing over the Internet or corporate intranet enables a user to make voice calls to associates and organizations around the world. NetMeeting voice conferencing offers many features, including half-duplex and full-duplex audio support for real-time conversations, automatic microphone sensitivity level setting to ensure that meeting participants hear each other clearly, and microphone muting, which lets users control the audio signal sent during a call. This voice conferencing supports network TCP/IP connections.

Support for the H.323 protocol enables interoperability between NetMeeting 2.0 and other H.323-compatible voice clients. The H.323 protocol supports the ITU G.711 and G.723 audio standards and Internet Engineering Task Force (IETF) RTP and RTCP specifications for controlling audio flow to improve voice quality. On MMX-enabled computers, NetMeeting uses the MMX-enabled voice codecs to improve performance for voice compression and decompression algorithms. This will result in lower CPU use and improved voice quality during a call.

With NetMeeting 2.0, a user can send and receive real-time visual images with another conference participant using any video for Windows-compatible equipment. They can share ideas and information face-to-face, and use the camera to instantly view items, such as hardware or devices, that the user chooses to display in front of the lens. Combined with the video and data capabilities of NetMeeting 2.0, a user can both see and hear the other conference participant, as well as share

information and applications. This H.323 standards-based video technology is also compliant with the H.261 and H.263 video codecs.

Multipoint data conferencing using T.120. Two or more users can communicate and collaborate as a group in real time. Participants can share applications, exchange information through a shared clipboard, transfer files, collaborate on a shared whiteboard, and use a text-based chat feature. Also, support for the T.120 data conferencing standard enables interoperability with other T.120-based products and services. The following features comprise multipoint data conferencing:

Application sharing: A user can share a program running on one computer with other participants in the conference. Participants can review the same data or information, and see the actions as the person sharing the application works on the program (for example, editing content or scrolling through information.) Participants can share Windows-based applications transparently without any special knowledge of the application capabilities.

The person sharing the application can choose to collaborate with other conference participants, and they can take turns editing or controlling the application. Only the person sharing the program needs to have the given application installed on their computer.

Shared Clipboard: The shared clipboard enables a user to exchange its contents with other participants in a conference using familiar cut, copy, and paste operations. For example, a participant can copy information from a local document and paste the contents into a shared application as part of a group collaboration.

File Transfer: With the file transfer capability, a user can send a file in the background to one or all of the conference participants. When one user drags a file into the main window, the file is automatically sent to each person in the conference; they can then accept or decline receipt. This file transfer capability is fully compliant with the T.127 standard.

Whiteboard: Multiple users can simultaneously collaborate using the whiteboard to review, create, and update graphic information. The whiteboard is object-oriented (versus pixel-oriented), enabling participants to manipulate the contents by clicking and dragging with the mouse. In addition, they can use a remote pointer or highlighting tool to point out specific contents or sections of shared pages.

Chat: A user can type text messages to share common ideas or topics with other conference participants, or record meeting notes and action items as part of a collaborative process. Also, participants in a conference can use chat to communicate in the absence of audio support. A “whisper” feature lets a user have a separate, private conversation with another person during a group chat session.

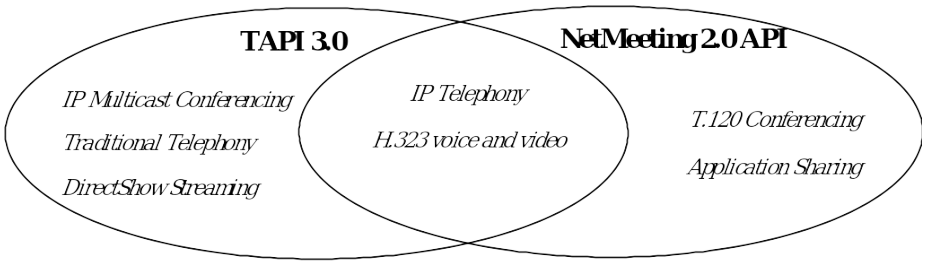
NetMeeting 2.0 Software Development Kit. This SDK enables developers to integrate this conferencing functionality directly into their applications or Web pages. This open development environment supports international communication and conferencing standards and enables interoperability with products and services from multiple vendors.

Also in the NetMeeting SDK are APIs to add non-standard codecs and to access ILS servers via LDAP, as well as an ActiveX™ control to simplify adding conferencing capabilities to Web pages.

For more information on the *Microsoft NetMeeting 2.0 Software Development Kit*, see http://www.microsoft.com/netmeeting/sdk.

Features of TAPI 3.0 and NetMeeting 2.0

TAPI 3.0 and NetMeeting 2.0 both support core IP Telephony capabilities. Each platform offers unique benefits: TAPI 3.0 seamlessly integrates traditional telephony with IP Telephony, providing a COM-based, protocol-independent call-control and data streaming infrastructure. NetMeeting 2.0 SDK supports T.120 conferencing and application sharing in addition to IP Telephony. Applications using TAPI 3.0 and the NetMeeting 2.0 API interoperate using H.323 audio and video conferencing.



When to Use TAPI 3.0 and NetMeeting 2.0

Because TAPI 3.0 and NetMeeting 2.0 both support core IP Telephony capabilities (including support for H.323), developers may want to consider the following guidelines when choosing an API for their IP Telephony applications:

TAPI 3.0. This is the API to use if you are doing IP Telephony in your application. TAPI 3.0 is especially valuable in the world of client/server computer telephony integration, for combining IP Telephony with traditional telephony, and for IP multicast of voice and video.

NetMeeting 2.0 API. This is the API to use if you are doing real-time collaboration and want to integrate voice, video, and data conferencing into your application. The NetMeeting API is useful for applications that want to integrate application sharing, whiteboard functionality, and

multipoint file transfer with voice and video sessions.